

# Internationalization

Welcome all translators. Please review Crowdin below. That is the easy way to contribute your skills to the development of OpenCPN and Plugins.

## Add a new Language

1. To translate OpenCPN into any language, download the **opencpn.pot** file from [here](#).
2. Next rename this file using the [ISO Alpha-2 Country Code](#) for your language and changing the extension .pot to .po, e.g. **opencpn\_it\_IT.po** for Italian.
3. Use [Poedit](#), or edit the file manually by entering values for the translated terms into the Msgstr “ ” fields.
4. Test the file in OpenCPN by selecting the appropriate language. On Linux, you may need to install further language libraries (locales) to be able to select all languages. Here's how to do so on Ubuntu based systems:

```
#To list already installed locales
locale -a

#To see a list of supported langs
less /usr/share/i18n/SUPPORTED

#To install a new locale
sudo locale-gen de_DE.UTF-8
#This is an example for German, Unicode...

#To update the list in the system
sudo update-locale
```

## To improve or extend any existing translation (local)

1. To improve or extend any existing translation, download the respective .po file, e.g. **opencpn\_it\_IT.po** for Italian.
2. Use [Poedit](#), or edit the file manually by entering values for the translated terms into the Msgstr “ ” fields.
3. Test the file in OpenCPN by selecting the appropriate language. On Linux, you may need to install further language libraries to be able to select all languages.

## Use Crowdin with Browser

With version 3.3.1824 we have changed OpenCPN's translations to be managed by [Crowdin](#).

There are a couple of reasons behind this decision:

- Some of the translations lack considerably behind and simply miss a maintainer
- A lot of the plug-ins lack the translations completely or almost completely
- It is hard for more people to cooperate on a language
- It needs effort from people who would like to contribute their time and language skills to install Poedit, learn how to use it and get the current PO/POT files, so the less geeky types may feel that it's too much for them
- It gets a bit messy at times to manage all the translations submitted through different channels and the time can be better spent on writing new code and fixing bugs

We have created a project on [Crowdin](#) (Many thanks to these guys for generously supporting open source projects) to maintain the localizations. From now on all that you need to help us keep the translations up to date is a web browser and a bit of time.

**Please join us by signing up at <https://crowdin.net/project/opencpn/invite>**

## Concerning "Items" and "Texts", the rule is:

1 Item in the language file 1 = Display text.

Of course, this requires you to put more items in this file, but this is the only way for each translator to take account of the peculiarities of its language.

Currently, all the other plugins and the heart of OpenCPN follow this rule and we can always translate very correctly.

## FAQ

***I have a question about translation priority between core and plugins, ....How can this be dealt with?***

It's simple. The language files of the plugins are loaded first. Then, the core language file is loaded. the whole thing is a single language file. If an item appears in a language file of a plugin and also, strictly identical, in the file of the core, this constitutes a single item. So the translation of the core takes the place of the translation of the plugin.

The solution, to avoid translation conflicts, is that plugin developers must ensure that the items they use are different from those of the core.

It is not necessary that the difference be very great. Only one different character is enough.

***How does ODraw ensure that it's translations are used?***

ODraw uses a process to make sure its translations for are used for all its items. It does not affect the

mainline translations, just the translation catalogue used by ODraw is its own. All plugins can use this method without impacting any other. The one downside is that you need to have translations for all words as the cascade back through the different catalogues may not give you the answer you expect. This is achieved by redefining the translation macro (done by cmake and compiler) such that the source code looks the same, but the catalogue being used is different.

This process is currently used in the **ODraw** and **Watchdog** plugins. It is a simple change to a couple of files to implement.

### ***For Compiling, what is the Process to update Internationalization files when compiling plugins?***

The “normal” routine is that when new development for each plugin is ready, at least for text parts, address a message to Pavel or me and we can upload the new strings, the POT-file, to Crowdin. When translators are ready or at least to an acceptable level and before release a new message will result in download of translations, the po-files, and a PR to the plugin's Github. The crucial task is to message when new strings are ready for Crowdin. “As simple as that”.

From:  
<https://opencpn.org/wiki/dokuwiki/> - **OpenCPN Manuals**

Permanent link:  
[https://opencpn.org/wiki/dokuwiki/doku.php?id=opencpn:developer\\_manual:developer\\_guide:internationalization](https://opencpn.org/wiki/dokuwiki/doku.php?id=opencpn:developer_manual:developer_guide:internationalization)

Last update: **2020/03/06 13:38**

